Claims

What is claimed is:

1 A method of organizing blocks of memory in a digital computer so as implement an associative memory that, for a given set of Boolean variables, maps Boolean-variable-to-Boolean-value assignments to values stored in the computer memory. Blocks of memory represent instances of class `CFLOBDD` (CFLOBDDs) and instances of class `Grouping` (proto-CFLOBDDs) according to the class definitions given in Figure 12 and Structural Invariants 1–5. The method comprises the following steps:

a. The blocks of memory are connected to form a structure that represents a hierarchically structured graph in which each matched path through the structure (i) corresponds to a unique Boolean-variable-to-Boolean-value assignment, and (ii) leads to an element of the computer memory in which is stored the piece of information associated with that Boolean-variable-to-Boolean-value assignment.

2 The method of claim **1**, wherein the connections between blocks of memory are established by the following steps:

a. Create a decision tree that represents the information to be stored in the associative memory, and whose height is an integral power of 2

b. Apply Algorithm 1 to form a multi-terminal CFLOBDD representation in memory.

3 A method for representing groupings and proto-CFLOBDDs in the memory of a computer so that equality of proto-CFLOBDDs can be tested in constant time, comprising the following steps:

a. Allocate a table in which to store the unique representatives of values of type `Grouping`.

b. Use the table to perform memoization during operations that construct values of type `Grouping` in the computer memory, so that only a single representative is ever constructed for each value of type `Grouping`.

c. Determine whether two values of type `Grouping` are equal (and hence whether two proto-CFLOBDDs are equal) by testing whether their addresses in the computer memory are equal.

4 A method for representing multi-terminal CFLOBDDs in the memory of a computer so that equality of multi-terminal CFLOBDDs can be tested in constant time, comprising the following steps:

a. Allocate a table in which to store unique representitives of values of type `CFLOBDD`.

b. Use the table to perform memoization during operations that construct values of type `CFLOBDD` in the computer memory, so that only a single representative is ever constructed for each value of type `CFLOBDD`.

c. Determine whether two values of type `CFLOBDD` are equal by testing whether their addresses in the computer memory are equal.

5 A method for obtaining, in the memory of a computer, a level-k CFLOBDD that represents a constant-valued function of the form $\lambda x_0, x_1, \ldots, x_{2^k-1}.v$, comprising the following steps:

a. Apply routine `ConstantCFLOBDD(k,v)` to form a CFLOBDD representation in memory.

6 A method for obtaining, in the memory of a computer, a level-k CFLOBDD that represents a Boolean-valued projection function of the form $\lambda x_0, x_1, \ldots, x_{2^k-1}.x_i$, where i ranges from 0 to $2^k - 1$, comprising the following steps:

a. Apply routine `ProjectionCFLOBDD(k,i)` to form a CFLOBDD representation in memory.

7 A method for obtaining, in the memory of a computer, a level-k CFLOBDD that represents a step function of the form

$$\lambda x_0, x_1, \ldots, x_{2^k-1}. \begin{cases} v1 & \text{if the number whose bits are } x_0 x_1 \ldots x_{2^k-1} \text{ is strictly less than } i \\ v2 & \text{if the number whose bits are } x_0 x_1 \ldots x_{2^k-1} \text{ is greater than or equal to } i \end{cases}$$

where i ranges from 0 to 2^{2^k}, comprising the following steps:

a. Apply routine `StepCFLOBDD(k,i,v1,v2)` to form a CFLOBDD representation in memory.

8 A method for obtaining a representation of a Boolean-valued CFLOBDD in the memory of a computer, comprising the following steps:

a. Apply routine `ComplementCFLOBDD(c)` on a given Boolean-valued CFLOBDD `c` to form a Boolean-valued CFLOBDD representation in memory.

9 A method for obtaining a representation of a multi-terminal CFLOBDD in the memory of a computer, comprising the following steps:

a. Apply routine `FlipValueTupleCFLOBDD(c)` on a given multi-terminal CFLOBDD `c` to form a multi-terminal CFLOBDD representation in memory.

10 A method for obtaining a representation of a multi-terminal CFLOBDD in the memory of a computer, comprising the following steps:

a. Apply routine `ScalarMultiplyCFLOBDD(c,v)` on a given multi-terminal CFLOBDD `c` and value `v` to form a multi-terminal CFLOBDD representation in memory.

11 A method for obtaining a representation of a Boolean-valued CFLOBDD in the memory of a computer, comprising the following steps:

a. Apply routine `BinaryApplyAndReduce` for evaluating binary Boolean operations on Boolean-valued CFLOBDDs to form a Boolean-valued CFLOBDD representation in memory.

12 A method for obtaining a representation of a multi-terminal CFLOBDD in the memory of a computer, comprising the following steps:

a. Apply routine `BinaryApplyAndReduce` for evaluating binary operations on multi-terminal CFLOBDDs to form a multi-terminal CFLOBDD representation in memory.

13 A method for obtaining a representation of a Boolean-valued CFLOBDD in the memory of a computer, comprising the following steps:

a. Apply routine `TernaryApplyAndReduce` for evaluating ternary Boolean operations on Boolean-valued CFLOBDDs to form a Boolean-valued CFLOBDD representation in memory.

14 A method for obtaining a representation of a Boolean-valued CFLOBDD in the memory of a computer, comprising the following steps:

a. Apply routine `TernaryApplyAndReduce` with the operation ITE as the fourth argument, and evaluate binary Boolean operations on Boolean-valued CFLOBDDs according to the table given in Figure 28 to form a Boolean-valued CFLOBDD representation in memory.

15 A method for obtaining a representation of a multi-terminal CFLOBDD in the memory of a computer, comprising the following steps:

a. Apply routine `TernaryApplyAndReduce` for evaluating ternary operations on multi-terminal CFLOBDDs to form a multi-terminal CFLOBDD representation in memory.

16 A method carried out by a computer for constructing a multi-terminal CFLOBDD that represents the Kronecker product of two matrices A and B, where the values of A are drawn from $\{w_0, \ldots, w_m\}$ and the values of B are drawn from $\{v_0, \ldots, v_n\}$, comprising the following steps:

a. Create representations of the matrices A and B as level-k multi-terminal CFLOBDDs in the computer memory, under the interleaved variable ordering. (Without loss of generality, we can make the assumption that the sequences of exit vertices in the two multi-terminal CFLOBDDs are mapped to the values $[w_0, \ldots, w_m]$ and $[v_0, \ldots, v_n]$, respectively.)

b. Create a level $k+1$ grouping that has $m+1$ middle vertices, corresponding to the values $[w_0, \ldots, w_m]$, and $(m+1)(n+1)$ exit vertices, corresponding to the values

$$[w_i v_j : i \in [0..m], j \in [0..n]].$$

c. For each middle vertex, which corresponds to some value w_i, for $0 \leq i \leq m$, create a B-connection to the proto-CFLOBDD of B, and a return tuple from the exit vertices of the proto-CFLOBDD of B to the exit vertices of the level $k+1$ grouping that correspond to the values $[w_i v_0, \ldots, w_i v_n]$.

d. If any of the values in the sequence

$$[w_i v_j : i \in [0..m], j \in [0..n]]$$

are duplicates, make an appropriate call on `Reduce` to fold together the classes of exit vertices that are associated with the same value, thereby creating a multi-terminal CFLOBDD in the computer memory.

17 A method carried out by a computer for constructing a multi-terminal CFLOBDD that represents the Reed-Muller transform matrix R_{2^j}, for $j \geq 0$, comprising the following steps:

a. Apply routine `ReedMullerCFLOBDD(j+1)` to form a multi-terminal CFLOBDD representation in memory.

18 A method carried out by a computer for constructing a multi-terminal CFLOBDD that represents the inverse Reed-Muller transform matrix S_{2^j}, for $j \geq 0$, comprising the following steps:

a. Apply routine `InverseReedMullerCFLOBDD(j+1)` to form a multi-terminal CFLOBDD representation in memory.

19 A method carried out by a computer for constructing a multi-terminal CFLOBDD that represents the Walsh transform matrix W_{2^j}, for $j \geq 0$, comprising the following steps:

a. Apply routine `WalshCFLOBDD(j+1)` to form a multi-terminal CFLOBDD representation in memory.

20 A method carried out by a computer for constructing a multi-terminal CFLOBDD that represents the Boolean Haar Wavelet transform matrix H_{2^j}, for $j \geq 0$, comprising the following steps:

a. Apply routine `HaarCFLOBDD(j+1)` to form a multi-terminal CFLOBDD representation in memory.

21 A method carried out by a computer for compressing data for subsequent storage and/or transmission of the data in compressed form, comprising the following steps:

a. If the length of the signal is s, the signal is padded to be of length 2^{2^k}, for the smallest value of k for which $s \leq 2^{2^k}$. For padding, a distinguished value is used that indicates that these elements are not part of the signal.

b. Building a CFLOBDD for which the values produced by evaluating successive Boolean-variable-to-Boolean-value assignments, considered in lexicographic order, match the padded signal.

22 A method carried out by a computer for compressing data for subsequent storage and/or transmission of the data in compressed form, comprising the following steps:

a. The signal to be compressed, consisting of a sequence of values drawn from some finite value space, is considered to be the values that label, in left-to-right order, the leaves of a decision tree.

b. If the length of the signal is s, the decision tree used is one whose height is 2^k, where k is the smallest value for which $s \leq 2^{2^k}$; the extra leaves are labeled with a distinguished value that indicates that these elements are not part of the signal.

c. Apply Algorithm 1 to form a multi-terminal CFLOBDD representation in memory.

23 A method carried out by a computer for uncompressing data that has been stored in compressed form as a CFLOBDD c, comprising the following steps:

a. Form the sequence in memory obtained by evaluating c for each Boolean-variable-to-Boolean-value assignment in lexicographic order, up until the first time that one of the distinguished values being used to indicate a value that is not part of the signal is obtained.

24 A method carried out by a computer for uncompressing data that has been stored in compressed form as a CFLOBDD c, comprising the following steps:

a. Apply routine `UncompressCFLOBDD` to c to form a sequence of values in memory.

b. The sequence of values is then processed to remove any occurrences of the distinguished value being used to indicate a value that is not part of the signal.

25 A method carried out by a computer for uncompressing data that has been stored in compressed form as a CFLOBDD c, comprising the following steps:

a. Apply a version of routine `UncompressCFLOBDD` to c, wherein processing is halted, and the sequence of values that has been formed in memory is returned as the answer, the moment that the first occurrence of one of the distinguished values being used to indicate a value that is not part of the signal is encountered at line [48] of routine `UncompressCFLOBDD`.